Utilising Standards Based Approaches to Information Sharing and Interoperability in Manufacturing Decision Support

Anne-Françoise Cutting-Decelle(*[1]), Line Pouchard(**), Jean-Jacques Michel(***), Robert Young(+), Bishnu Das(+)

(*)University of Evry / IUT-Dept Organisation and Production Management, 91025 EVRY, France

(**)Computer Science and Mathematics, Oak Ridge National Laboratory, Oak Ridge, TN 37831-6367

(***)Idpiconseil, 94700 MAISONS ALFORT, France

(+)Wolfson School of Mechanical and Manufacturing Engineering, University of Loughborough (UK)

ABSTRACT

The need for high quality information to support the improvement of the production process is well recognised, as is the importance of sharing information in order to reduce production costs. The use of IT systems in manufacture has huge potential to support this need. However, for information sharing through software systems to be fully effective it is essential that such systems can easily interoperate. Given the range of systems which manufacturing businesses need to use and the complexity of business interactions in extended enterprises there is an important role for standards to play in providing common tools and languages by which information sharing and interoperability can be achieved. This paper discusses the issues involved in advancing the levels of information sharing and interoperability which can be achieved in manufacturing software support systems and the role which standards should play. It highlights the use of manufacturing information sharing standards, especially those being developed in the ISO Industrial Data Committee such as through the MANDATE and PSL groups. It goes on to discuss research which is exploring the use of PSL to provide a higher level of interoperability through knowledge sharing.

1. INTRODUCTION

The term “manufacturing interoperability” refers to the ability to share technical and business information seamlessly throughout an extended manufacturing enterprise (supply chain) [1]. This information, previously shared in a variety of ways including paper and telephone conversations, must now be passed electronically and error-free with suppliers and customers around the world. A study, achieved by the NIST in 2002 [1] was aimed at identifying the economic impact of the use of standards in industry, particularly the ISO 10303 STEP standard with the objective of conducting an economic impact assessment of STEP’s use by transportation equipment industries, namely the automotive, aerospace, shipbuilding, and specialty tool and die industries. Both the full potential and current realized benefits are quantified. In addition, the study investigates the impact of NIST’s administrative and technical contributions to STEP. The authors of the study estimate the economic value of the efficiency gains due to improved data exchange enabled by using STEP, and quantify NIST’s contributions to those gains. Data collected

[1] Corresponding author : Tel. : (33) 169 47 73 11, Fax : (33) 169 47 73 06, Email : afcd@skynet.be

from industry surveys and case studies are used to estimate the potential benefits of existing STEP capabilities. They estimate that STEP has the potential of save $928 million (in 2001) per year by reducing interoperability problems in the automotive, aerospace, and shipbuilding industries. Currently approximately 17 percent ($156 million) of the potential benefits of STEP quantified within the scope of this study are being realized. A previous study commissioned by NIST [3] in 1999, had reported that the U.S. automotive sector alone expended one billion dollars per year to resolve interoperability problems. The study also reported that as much as 50% of this expenditure is attributed to dealing with data file exchange issues.

There are three principal approaches to reduce these exorbitant costs. The first is a point-to-point customized solution, which can be achieved by contracting the services of systems integrators. This approach is expensive since each pair of systems needs a dedicated solution. A second approach, adopted in some large supply chains, obliges all supply chain partners to conform to a particular solution. This approach does not solve the interoperability problem since the first or sub-tier suppliers are forced to purchase and maintain multiple, redundant systems. The third approach involves neutral, open, published standards. By adopting open standards the combinatorial problems is reduced from n^2 to n, with bi-directional translators. Published standards also offer some stability in the representation they propose of the information models, an essential property for long-term data archiving. This paper highlights some of the standards developed within the ISO TC184 "Industrial Automation Systems and Integration" Committee, particularly those related to the manufacturing domain.

But the problem is far from solved. Interoperability standards are used in layers, from the cables and connectors, through networking standards, to the application or content standards such as those mentioned here, that is STEP, P-LIB, MANDATE and PSL. All of these layers must function correctly for interoperability to be achieved. The greatest challenges remain at the top of this stack of standards, in order to make them inter-operable. Due to the auto-enricheness capacities of the PSL language (through its ontology), this language can be considered as a powerful tool of this interoperability, enabling, for a near future, the consideration of a "universal interfacing".

The first part of this paper presents the environment of the ISO TC184 international standardisation committee, and some of the main standards developed by its different sub-committees. Then, we focus on standards belonging to the domain of manufacturing, notably the ISO 15531 MANDATE standard and on the links existing between the different standards. Another, and new approach for the TC184 committee about manufacturing interoperability appears in the work done within the SC5 committee ("Architcture, communications and integration frameworks") in the domain of enterprise modelling and engineering [4]. The ISO 18629 PSL language is developed jointly by the SC4 ("Industrial data") and the SC5 committees. This standard is described, then analysed in terms of the benefits that can be expected from its introduction in the exchanges of manufacturing information.

2. THE ISO STANDARDS IN THE DOMAIN OF INDUSTRIAL COMMUNICATIONS

2.1 SCOPE OF THE ISO TC184 INTERNATIONAL STANDARDISATION COMMITTEE

The ISO TC184 is one of the one hundred eighty eight committees member of the ISO (International Standardisation Organisation, Geneva, CH). Its scope is : "*Standardisation in the field of industrial automation and integration concerning discrete part manufacturing and encompassing the applications of multiple technologies, i.e. information systems, machines and equipments and telecommunications*" [5]. Are excluded from the scope the following domains : electrical and electronic equipment (dealt with by the IEC/TC44) and programmable logical controllers for general applications (IEC/TC65). The scope of the committee means that the standards developed are : applicable to manufacturing and process industries, applicable to all sizes of business, applicable to extending exchanges across the globe through e-business.

2.2 DOMAINS COVERED BY THE STANDARDS

The standards developed within the ISO TC184 cover various domains related to industrial automation and integration, among which : enterprise modelling, enterprise architecture, communications and processes, integration of industrial data for exchange, access and sharing, life cycle data for process plants, manufacturing management, mechanical interfaces and programming methods, part libraries, physical device control, process specification language, product data, and robots for manufacturing environment. A list of some of the main (among the most recent) standards is provided in the Table 1.

Table 1 : List of some of the main (and most recent) ISO TC184 standards [6]

Number-year	Name
ISO 3592: 2000	Numerical control of machines - NC processor output - File structure and language format
ISO 4343: 2000	Numerical control of machines - NC processor output - Post processor commands
ISO 9506: 2003	Industrial automation systems and integration - Manufacturing Message Specification
ISO 9787: 1999	Manipulating industrial robots - Coordinate systems and motion nomenclature
ISO 9946: 1999	Manipulating industrial robots - Presentation of characteristics
ISO 10303: 1994	**Industrial automation systems and integration - Product data representation and exchange**
ISO 13281: 1997	Industrial automation systems and integration – Manufacturing Automation Programming Environment (MAPLE)
ISO 13584: 2001	Industrial automation systems and integration – Parts Library
ISO 14258: 1998	Industrial automation systems and integration – Concepts and rules for enterprise models
ISO 14649: 2003	Industrial automation systems and integration – Data model for computerized numerical controllers
ISO 15187: 2000	Industrial automation systems and integration - Manipulating industrial robots - Graphical user interfaces for programming and operation of robots (GUI-R)
ISO 15531: 2003	**Industrial automation systems and integration – Industrial manufacturing management data**
ISO 15704: 2000	Industrial automation systems – Requirements for enterprise-reference architectures and methodologies
ISO 15745: 2003	Industrial automation systems and integration – Open systems application integration framework
ISO 16100: 2002	Industrial automation systems and integration – Manufacturing software capability profiling for interoperability
ISO 18629: 2003	**Industrial automation systems and integration – Process specification language**
ISO/EN19439: 2004	Industrial automation systems and integration – Framework for enterprise modelling
ISO/EN19440: 2004	Industrial automation systems and integration – Constructs for enterprise modelling
IEC 62264: 2003	Industrial automation systems and integration – Enterprise control system integration

It is important to notice that all the standards of the list are not yet fully developed (particularly when they are made of multiple parts), on the other hand, the ISO/IEC 62264 standard is developed by the ISO/IEC SC65A joint committee. Besides, and not given in the list since it is only a TR (Technical Report) and not a standard, let us mention the ISO TR 10314, whose title is "Referencemodel for shop floor production standards, Part 1 – Reference model for standardisation, methodology for identification of requirements" [7].

This paper mainly focuses on the MANDATE and PSL standards, given their implication in manufacturing systems.

One of the most important standard of the list, known as an example of a successful open standard is ISO 10303 [8], informally known as STEP, the STandard for the Exchange of Product Model data. STEP, which is actually a family of standards, defines a neutral representation for product data over its entire life cycle. In the presentation of the MANDATE standard, we will highlight the numerous relationships existing between the two standards.

3. EXCHANGE OF MANUFACTURING INFORMATION USING STANDARDS : THE EXAMPLE OF THE ISO 15531 MANDATE STANDARD

3.1 SCOPE AND DESCRIPTION OF THE ISO 15531 MANDATE STANDARD

A manufacturing management system manages the flow of information, materials and products through the whole production chain, from suppliers, through manufacturers, assemblers, to distributors [9]. The relations among them may be structured in an electronic form to facilitate electronic exchanges. The production planning functions within the supplier plants are assumed to have strong relationships with the master production scheduling people of the main plant, who share with them information on the likely pattern of the future demands to allow suppliers to plan in turn their production. Three main categories of data related to manufacturing management can be distinguished : information related to the external exchanges (between main plant and suppliers), information related to the management of the resources used during the manufacturing processes and information related to the management of the manufacturing flows. Following this description, the standard is divided into three series of parts, strongly related in order to preserve the consistency of the whole standard. The different parts make use of the EXPRESS language, written within the framework of the STEP project [10].

- Parts 15531-2x series (Production data for external exchanges) : work currently on stand-by, waiting for the developments of the Basic Semantic Repository [11].

- Parts 15531-3x series (Manufacturing resources usage management data) : addresses resource usage management, operation management, with quality, maintenance and safety features. Different aspects are considered about the resources : their description, the way of using and maintaining them (in close relationship to the production) ; the description of the activities, operations and functions a resource achieves (capacity and capability) ; the model of information needed to define, operate and monitor the resource. Capacities and capabilities of the resources are modelled at a very generic level, enabling its use to develop more precise resources models aimed at specific industrial activities, or specific functions [12].

- Parts 15531-4x series (Manufacturing flow management data) : supports the control and the monitoring of the flows in manufacturing or industrial processes, with close links to resources usage management data. On the basis of existing standards for process plans, the standard enables the description of : information and materials flows in industrial manufacturing processes ; all the information necessary for scheduling, controlling and monitoring the flows. This series of part provides a "Time model" [13] enabling a representation of concepts related to the time, as they mainly appear in the scheduling function, and a "Manufacturing flow management data model" [14].

3.2 JOINT USE OF MANDATE WITH OTHER MANUFACTURING RELATED STANDARDS (STEP, P-LIB, …)

MANDATE addresses operations dealing with product manufacturing (ISO 10303 STEP), and makes use of components descriptions, as they are defined in the ISO 13584 P-LIB [15] libraries. The links between the different standards clearly appears on the model of the part 43, "manufacturing flow management data schema" represented in the Fig. 1. On this schema, where some of the links to other standards have also been represented, it is easy to notice that the data exchanges during the production cycle are strongly related to the product itself, the components the product is made of, the resources needed, but also to the time and the different flows circulating.

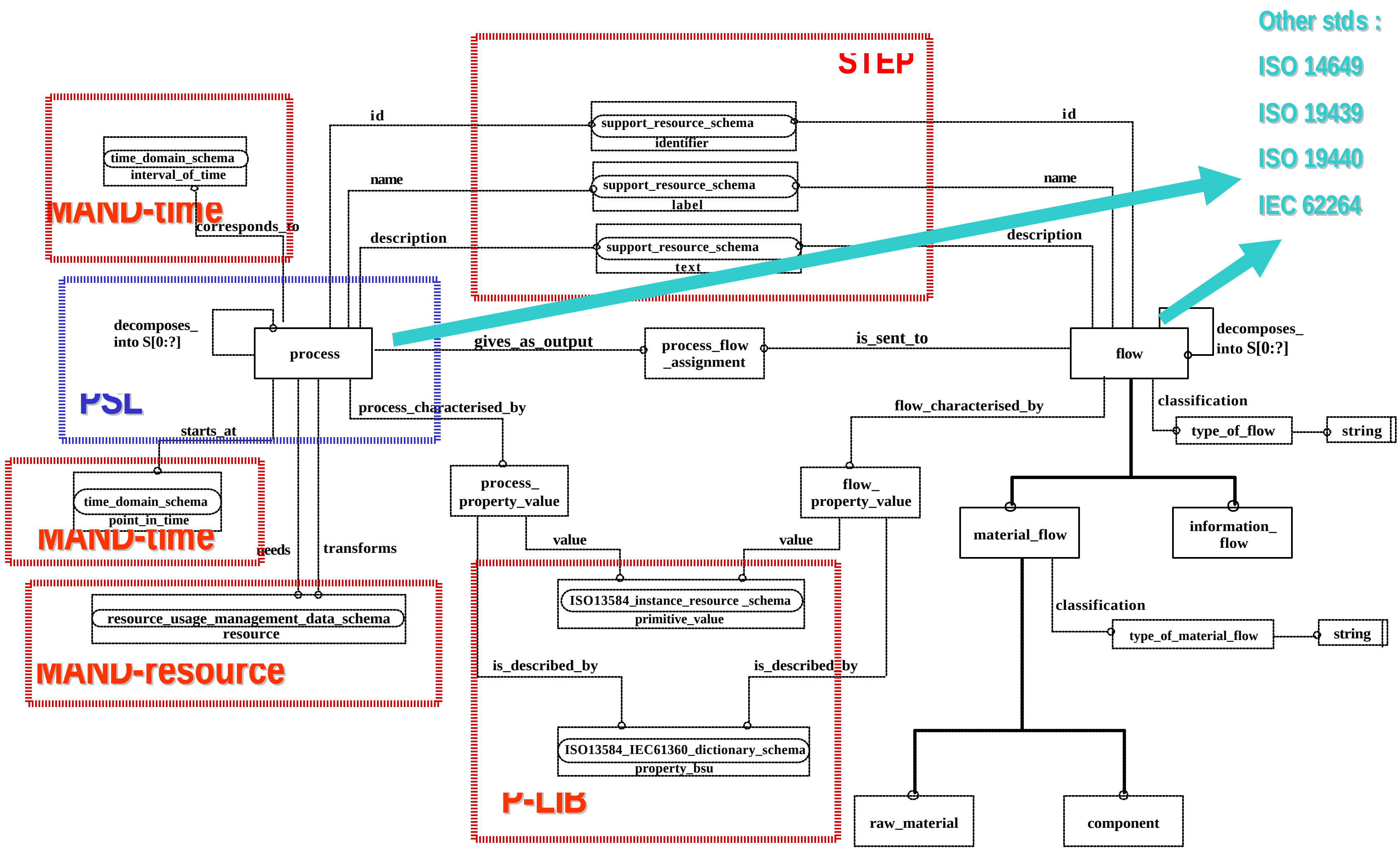

Fig. 1 : MANDATE part 43 model : manufacturing flow management data schema (EXPRESS-G representation) [29]

3.3 PROBLEMS ARISING

All those models, but also all the software applications developed on the basis of those models handle a great number of concepts, of terms described using given vocabularies. Some of those terms are ambiguous, since their meaning can differ according to the partner, and/or the application involved : in the Figure 2, what is called a "resource" for the application_A is a *machine*, once expressed in a common language, but for the application_B, a resource will be made of the *materials* used : in that case, it is very difficult to be able to identify the common concepts hidden behind the use of different terms !!

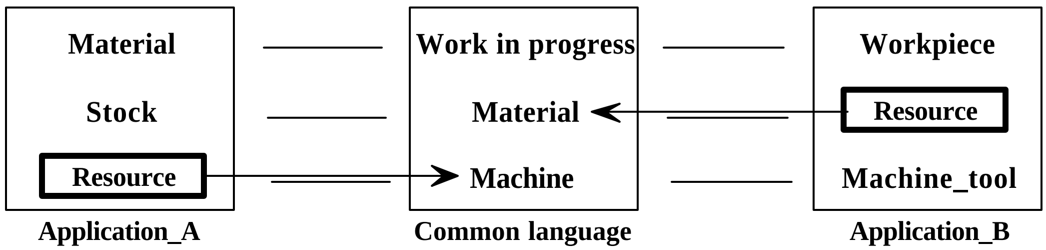

Figure 2 : semantic ambiguïty in the representation of manufacturing information [16]

This problem of semantic ambiguïty is one of the most important features, hindering an "intelligent" interoperability among software applications. In the following section, we present the use of the PSL language, as a way of overcoming this problem. An example is provided by the Figure 1, within the rectangle called "STEP" : the **flow** and **process** entities of the part 43 of MANDATE make use of entities (provided by the part 41 of STEP) coming from the STEP schema called "**support_resource_schema**". Within the name of this schema, ***resource*** has to be taken with the meaning of a "construct", that is a set of structured entities, thus avoiding to re-write them. This meaning is completely different from the meaning of the term "**resource**" as it is used in the series 3x.

4. A HIGH LEVEL INTEROPERABILITY IN MANUFACTURING BASED ON KNOWLEDGE SHARING : THE ISO 18629 PSL LANGUAGE

4.1 PRESENTATION OF THE ISO 18629 PSL STANDARD

The development of the ISO 18629 PSL language started at the National Institute of Standards and Technology (NIST, USA). The standard currently under development is aimed at creating a neutral, high-level language for specifying processes and the integration of multiple process-related applications throughout the manufacturing life cycle [17]. This language is also formal and based on first-order logic, a mathematical set theory [18]. The scope is limited to the realm of discrete processes related to manufacturing, including all processes in the design/manufacturing life cycle.

4.1 THE LANGUAGE

As a formal language, PSL is a lexicon (set of logical and non-logical symbols) and a grammar (specification of how these symbols can be combined to make well-formed formulae). All are chosen to represent the basic concepts in the PSL ontology. The underlying grammar used for PSL is roughly based on KIF (Knowledge Interchange Format) [19], formal language based on first-order logic developed for the exchange of knowledge among computer programs with disparate representations. To date, PSL contains more than 330 concepts and 46 definitional extensions (see Table 2).

4.2 THE ONTOLOGY

The foundation of the process specification language is the PSL ontology, which provides rigorous and unambiguous definitions of the concepts necessary for specifying manufacturing processes to enable the exchange of process information. The PSL ontology is essentially two-tiered. The foundation of the ontology is a set of process-related concepts that are common to ALL manufacturing applications. These concepts constitute the core of the PSL ontology and include concepts such as objects, activities, activity occurrences, and timepoints. Since these concepts, alone, only allow for the exchange of very simple process specifications, the ontology includes a mechanism to allow for extensions to these core concepts (the second tier) to ensure the robustness of the ontology.

4.3 ORGANISATION OF THE STANDARD

To date, the standard is made of the following components : see Table 2 [17]

Table 2 : organisation of the ISO 18629 PSL standard

Series	Number	Name
Core theories	IS 18629-1	Overview and basic principles [17]
	ISO DIS 18629-11	PSL-Core [20]
	ISO CD 18629-12	Outer Core [21]
	ISO CD 18629-13	Duration and ordering theories [22]
Definitional extensions	ISO CD 18629-41	Definitional extension : activity extensions [23]
	ISO WD 18629-42	Definitional extension : temporal and state extensions [24]
	ISO WD 18629-43	Definitional extension : activity ordering and duration extensions

The language is based on the following concepts :

- PSL Core : set of axioms written in the basic language of PSL. There are four primitive classes, two primitive functions, and seven primitive relations in the ontology of PSL Core. The classes are `object, activity, activity_occurrence` and `timepoint`. The relations are `before, occurrence_of, participates_in, between, beforeEq, betweenEq, exists_at, is_occurring_at`. The functions are `beginof, endof`. `object, activity, activity_occurrence` and `timepoint`s (or POINTs for short), are known collectively as entities, or things. These classes are all pairwise disjointed. The PSL Core is provided with a list of 17 fundamental axioms ;

- Foundational Theories : set of theories whose expressive power is sufficient for giving precise definitions of, or axiomatizations for, the primitive concepts of PSL, thus greatly enhancing the precision of semantic translations between different schemes. The current theories are : Outer Core, Duration and ordering theories, Resource theories, Actor and agents theories ;

- Extensions : third component of PSL, the extension provides the resources to express information involving concepts that are not part of PSL Core. To define an extension, new constants and/or predicates are added to the basic PSL language, and, for each new linguistic item, one or more axioms are given that constrain its interpretation. In this way one provides a semantics for the new linguistic items. The current extensions deal with Activity, Temporal and state, Activity ordering and duration, Resource roles, Resource sets, Processor activities.

4.4 USE OF THE STANDARD FOR INTEROPERABILITY : EXAMPLE

In practice, the extensions are what software products claiming compliance with PSL will utilize. PSL Core and Foundational Theories ensure that a formal basis for the extensions exist. But practical use rely on extensions. In the case where an application or a new design/life cycle domain requires designing new extensions, the new extensions will have to be expressed and proven using PSL Core and foundational theories to be PSL compliant. In that sense, the PSL ontology is extensible.

The use of PSL to develop a high level translation between two applications (Applications A and B) is explored in this example. Figure 8 below illustrates the complete transaction between applications A, B, and PSL. This translation is a three-stage process, with a syntactic translation and a semantic translation detailed in Figures 3-7 [16] :

- Syntactic translation : the native syntax of an application is expressed PSL syntax (KIF) via a parser. This parser keeps the terminology of the application intact.

- Semantic translation to PSL : keeping the KIF syntax for the terminology of the application of interest, KIF definitions are written for that application using PSL definitions. These definitions are found within the concepts of the PSL extensions. A question wizard is available to facilitate the attribution of definitions to the terminology and concepts of an application to PSL definitions. These translation definitions between an application ontology and PSL can be derived from the ontological definitions and axioms provided in the different parts of the language.

- Semantic translation from one application to another : once the steps above have been completed for each applications, the terminology of the source application is mapped to that of the target application using PSL as the intermediate language.

The two applications are called ApplicationA and ApplicationB. In this example, Application A's original syntax and terminology reads:**{resourceA: inject_mold (x)}.** First the concepts of application A must be formally represented using KIF syntax or any other formal representation that can be translated to KIF. The concepts of application A represented in KIF syntax express that inject_mold is a resource (Figure 3):

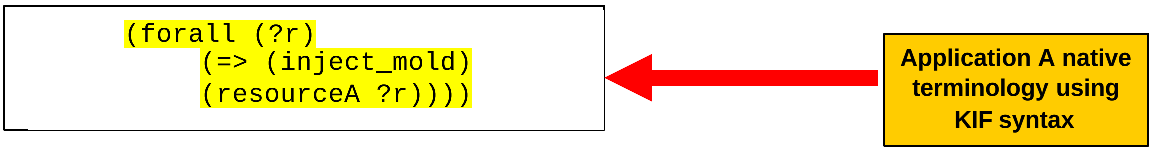

Fig. 3 : KIF

The ontology for Application A is also expressed. This non-trivial task is performed using technical documentation that is supplemented by interviews with developers and training by vendors. The question wizard is also of help for this task. Expressing the ontology of application A means expressing what kind of resource is implied by *resourceA* in application A terminology. Here *resourceA* is re-usable; it is a machine that can still be used by a process after another process that also requires resourceA completes its occurrence. The ontology of application A expresses the definition for *resourceA*, where ?r is the resource variable and ?a the process variable (Figure 4):

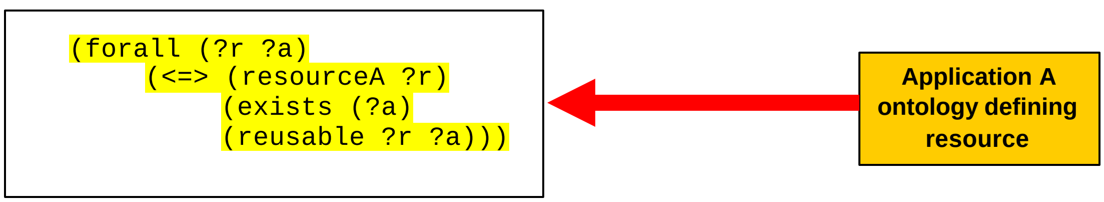

Fig. 4 : Ontology

Once a definition of resource is provided for application A, using Application A terminology, the next step is to look for possible mappings between *resourceA* and PSL concepts. 3 cases occur: 1)a one-to-one mapping is possible, 2)a one-to-one mapping is possible under certain conditions, 3)PSL does not contain the concept in question and needs to be extended to accommodate application A. For practical purposes only examples for Case 1 and 2 are given here. This mapping is done for every concept and relation contained in ApplicationA.

Case 1) PSL contains a concept of resource defined as -- a resource is any object that is required by some activity -- where "activity" and "requires" are defined elsewhere in PSL. This definition is expressed with KIF syntax:

```
(defrelation resource (?r) :=
    (and  (object ?r)
          (exists (?a)
          (requires ?a ?r))))
```

PSL also specifies roles for resources and defines the concept of a reusable resource as -- a resource ?r is reusable by an activity ?a if any other activity that also requires ?r is still possible to perform after ?a completes its occurrence, in every possible future -- where "common,""occurs_over," "legal_interval," "legal," and "legal_activity" are defined elsewhere in PSL (Figure 5). This definition is expressed with KIF syntax (Figure 5):

```
(defrelation reusable (?r ?a1) :=
(forall (?a2 ?occ)
     (=>        (and (common ?a1 ?a2 ?r)
                (occurs_over ?a1 ?occ))
     (forall (?b)
          (=> (forall (?s3)
                (=> (and
                (legal_interval ?b)
          (situation_during ?s3 ?b)
          (occurs_during ?occ (legal ?s3)))
          (legal_activity ?a2?s3)))))))
```

PSL definition for a reusable resource

Fig. 5 : PSL

In our example, a mapping appears to exist, but more information about ApplicationA is needed before deciding for a one-to-one mapping. This information must be obtained from the documentation for Application A, and other sources (vendors). PSL specifies that a reusable resource is such that, as soon as one activity occurs, it is always possible to perform the next activity. An example of reusable resource according to PSL is a machine that does not require setup between activities. If the resource in application A satisfies this condition, we have a one-to-one mapping between the Application A concept, *resourceA*, and the PSL concepts, *resource* and *reusable*. This mapping is expressed as (Figure 6) :

```
      (forall (?r)
      (<=> (resourceA ?r)
            (and (resource ?r)
                  (reusable ?r)
                        (reusable
?r))))
```

Application A expressed using PSL concepts

Fig. 6 : Mapping

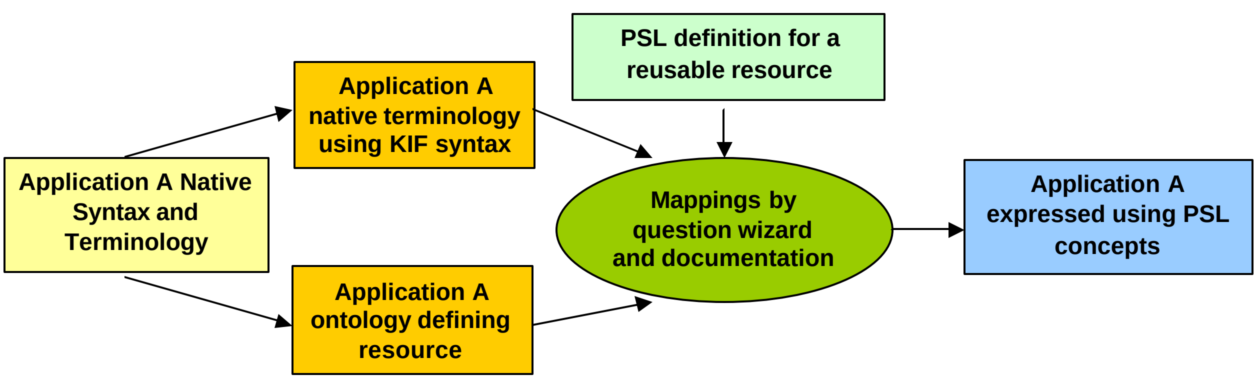

Figure 7 shows the workflow steps required for the complete mapping between Application A and PSL.

Similar steps are followed in the translation of Application B's concepts into PSL : an ontology expresses the concepts of Application B, the syntax of Application B is expressed in KIF, and a translator is written for mapping the concepts of Application B to PSL concepts. The result of the second mapping is that every concept of application B is defined in terms of PSL concepts. In addition, a index of each PSL concept contained in Application B is designed. (Figure 8). Our two applications A and B have now become "PSL compliant". When an engineer implements a model for a manufacturing task using Application A, the model is represented with PSL concepts, using a translator that is parsing the application to KIF. The model represented with PSL concepts is subsequently imported into Application B.

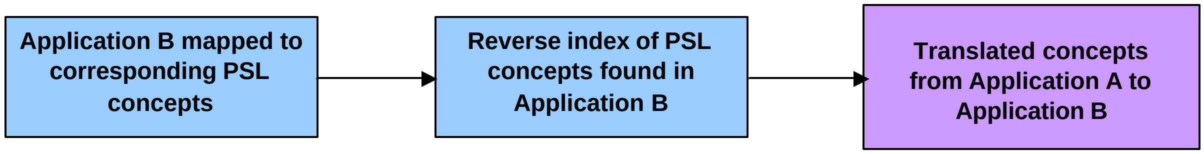

Figure 8: Completed translation

Case 2) Another possibility is a non-direct, conditional mapping between PSL and a given application. An example occured in the first pilot implementation with the ILOG (TM) scheduler offering the concept *ilcActivity*. The *ilcActivity* concept is narrower than the PSL concept of *activity*. How narrow must be defined for *ilcActivity* using PSL constraints. *IlcActivity* maps to the PSL *activity* concept further defined as a non-deterministic activity applied to a set of resources. The PSL definition is: -- An activity is a nondeterministic resource activity if and only if the reason that the activity is nondeterministic is because it is a nondeterministic selection activity with respect to some resource set. The one-to-one mapping with conditions between Application A concepts and PSL concepts is expressed:

```
(forall (?a)
    (=> (and (nondet_res_activity ?a)
        (primitive ?a))
        (<=> (ilcActivity ?a)
            (activity ?a))))
```

Other research work has been done or is currently on-going, showing examples or interoperability among software tools using PSL, notably at the University of Stanford (CIFE) [25], and at the University of Loughborough [26], [27].

4.5 BENEFITS OF THE USE OF PSL IN TERMS OF INTEROPERABILITY IN MANUFACTURING

In terms of interoperability in manufacturing systems, the benefits that can be expected from the use of the ISO 18629 PSL standard are very important.

If we take the exemple of the representation of the time, according to the two standards MANDATE and STEP :

- in MANDATE : the concept of « time » is very generic, enabling considering various types of time domains : continuous, discrete, or both. A specificity of this entity in ISO 15531 is that the time is considered independently of any event : a time « object » does not need to be related to a given triggered event to exist.

- in STEP: the time is necessarily related to an event since time consideration is necessary to sequence the triggering of an event.

Hence a direct translation from MANDATE.time to and from STEP.time is not necessarily as obvious as it may appear at a first glance : in that case, PSL can be helpful to identify common concepts behing entities whose name, or use, or meaning is not well defined enough to be sure that they share an identical semantics [28].

6. SUMMARY

This paper was aimed at showing to what extent standards based approaches can be helpful to facilitate information sharing and interoperability among software applications commonly used in manufacturing, and in manufacturing management. Most of the time, technical terms handled by those applications look similar, or, even worse, are exactly the same – however their meaning is different. This is particularly true for most of the standards mentioned in this paper, “built” more or less on the same “manufacturing-flavoured” vocabulary, but with very different and multiple interpretations of the same terms. Given its properties, and its structure, PSL can be considered as a powerful interoperability “tool” for the information systems of the enterprises.

REFERENCES

[1] *S.R. Ray, A.T. Jones, “Manufacturing interoperability”, CE: the vision for the future generation in research and applications, R. Jardim-Gonçalves et al. (Eds),Swedts & Zeitlinger, Lisse, ISBN 90 5809 622X, 2003*

[2] *NIST,"Economic impact assessment of the international standard for the exchange of product model data (STEP) in transportation equipment industries", Planning report #02-5, 2002*

[3] *NIST, "Interoperability cost analysis of the US automotive supply chain", Planning report #99-1, 1999*

[4] *D. Chen, F. Vernadat, "Standards on enterprise integration and engineering, a state-of-the-art", IJCIM Journal, Volume 17, n°3, April-May 2004*

[5] *http://www.iso.ch*

[6] *J.M. Chatelard, "ISO TC184 standard symposium", Symposium des normes du TC184, AFNOR, Paris, 2003*

[7] *ISO TR 10314, "Reference model for shop floor production standards, Part 1 – reference model for standardisation, methodology for identification of requirements", 1991*

[8] *ISO IS 10303-1, "Industrial automation systems and integration – product data representation and exchange – part 1 : overview and fundamental principles", 1994*

[9] *ISO IS 15531-1, "Industrial automation systems and integration – industrial manufacturing management data – part 1 : General overview", 2003*

[10] *ISO FDIS 10303-11, "Industrial automation systems and integration -- product data representation and exchange – part 11 : the EXPRESS language reference manual", 2000*

[11] *ISO PDTR 16668, "Basic Semantic Register (BSR) - Content (semantic components, semantic constructs and bridges)", ISO TC184/WG1, N007, 1999*

[12] *ISO DIS 15531-32, "Industrial automation systems and integration – industrial manufacturing management data – Resource usage management data : part 32 : Conceptual model for resource usage management data", 2003*

[13] *ISO CD 15531-42, "Industrial automation systems and integration – industrial manufacturing management data : Part 42 : Time model, 2003*

[14] *ISO WD 15531-43, "Industrial automation systems and integration – industrial manufacturing management data – Manufacturing flow management data – Part 43 : Conceptual model for manufacturing flow management", 2004*

[15] *ISO DIS 13584-1, "Industrial automation systems and integration – Parts Library – Part 1 : Overview and fundamental principles, 1997*

[16] *LC Pouchard, N. Ivezic, C. Schlenoff, «Ontology engineering for distributed collaboration in manufacturing », AIS Conference, 2000*

[17] *ISO IS 18629-1, "Industrial automation systems and integration – Process specification language – Part 1 : Overview and basic principles", 2004*

[18] *J. Barwise, J. Etchemendy, "The language of First-Order logic", 3rd edition, CSLI lecture notes, 1999*

[19] *M. Genesereth, R.., Fikes, "Knowledge Interchange Format (Version 3.0) - Reference Manual", Computer Science Dept., Stanford University, Stanford, 1992*

[20] *ISO DIS 18629-11, "Industrial automation systems and integration – Process specification language – Part 11 : PSL-Core", 2003*

[21] *ISO CD 18629-12, "Industrial automation systems and integration – Process specification language – Part 12 : Outer Core", 2003*

[22] *ISO WD 18629-13, "Industrial automation systems and integration – Process specification language – Part 13 : Duration and ordering theories", 2003*

[23] *ISO CD 18629-41, "Industrial automation systems and integration – Process specification language – Definitional extensions: Part 41 : Activity extensions", 2003*

[24] *ISO WD 18629-42, "Industrial automation systems and integration – Process specification language – Part 42 : temporal and state extensions", 2003*

[25] *K.H. Law,"Process specification and simulation",PSL quaterly progress report, Stanford University, CIFE, 2001*

[26] *G. Tesfagaber, A.F. Cutting-Decelle, C.J. Anumba, A.N. Baldwin, N.M. Bouchlaghem, "Semantic process modelling for applications integration in AEC", International workshop on informatin technology in civil Engineering, ASCE, 2002*

[27] *A.F. Cutting-Decelle, R.I.M. Young, B.P. Das, C.J. Anumba, A.N. Baldwin, N.M. Bouchlaghem, "A multi-disciplinary representation of the supply chain information in construction : an innvoative approach to project management", TMCE Conference, 2004*

[28] *J.J. Michel, A.F. Cutting-Decelle, "Relationships between manufacturing related standards", WG8 contribution, 2000*

[29] *J.J. Michel, A.F. Cutting-Decelle, "Interoperability in manufacturing : the ISO approach", Wolfson School Research Seminar, University of Loughborough, 2004*

The submitted manuscript has been authored by the US Department of Energy, Office **of Science of the Oak Ridge National Laboratory, managed for the U. S. DOE by UT-Battelle, LLC, under contract No. DE-AC05-00OR22725.** Accordingly, the U.S. Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.